UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/967,031	09/27/2001	Lokpraveen B. Mosur	2207/11305	4804

25693 7590 05/25/2004

KENYON & KENYON (SAN JOSE)
333 WEST SAN CARLOS ST.
SUITE 600
SAN JOSE, CA 95110

EXAMINER	
MCLEAN MAYO, KIMBERLY N	
ART UNIT	**PAPER NUMBER**
2187	18

DATE MAILED: 05/25/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/967,031	MOSUR ET AL.	
	Examiner	Art Unit	
	Kimberly N. McLean-Mayo	2187	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *08 March 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The enclosed detailed action is in response to the Amendment submitted on March 8, 2004.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claim 1-4, 7-12, 14, 16-20, 25-27 and 29-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over McDermott (USPN: 5,860,105) in view of Masubuchi et al. (USPN: 6,490,657).

Regarding claim 1, McDermott discloses an apparatus for cache flushing comprising a list structure for tracking the status of a plurality of cache entries, wherein the list structure does not contain cache data or addresses (Figure 4, comprised of References 331, 332); a query mechanism (scanning logic) for checking the list structure for the state of a cache entry (C 9, L 33-35; C 10, L 11-32); and a cache flush mechanism, logically coupled to the list structure and the cache, for flushing a cache entry and for modifying (by clearing the NDIRTY bits) the list structure to reflect the flushed state (C 10, L 2-8). McDermott does not explicitly disclose the list structure located outside the cache. However, Masubuchi discloses a list structure located outside of a cache (Figure 1, the list structure is comprised of References 32, 33, and 34). This feature taught by Masubuchi reduces logic in the cache, which provides space for

other/additional logic or otherwise allows the size of the cache to be reduced. Additionally, this feature relieves the cache from processing cache flushes, thereby allowing the cache to perform other tasks. Hence, it would have been obvious to one of ordinary skill in the art to locate the list structure in McDermott's system outside of the cache to reduce the logic in the cache and to allow external processing of the list structure to alleviate processing operations of the cache.

Regarding claim 2, McDermott and Masubuchi disclose the list structure comprising one bit per cache line (McDermott - Figure 4, Reference 331 – there is one bit per cache line for each cache line represented in the list structure).

Regarding claims 3-4, McDermott and Masubuchi disclose the list structure comprising one bit per plurality of cache lines (plurality of cache lines per way/set)(McDermott - Figure 4, Reference 332; C 9, L 35-37).

Regarding claim 7, McDermott and Masubuchi disclose a logical arrangement of the list structure matching the architecture of a cache (McDermott - C 9, L 39-41).

Regarding claim 8, McDermott and Masubuchi disclose the cache flush mechanism modifying a cache state responsive to the results of a query of the list structure (McDermott – C 10, L 2-8).

Regarding claims 9-10, McDermott and Masubuchi do not disclose the cache flush mechanism logically coupled to a higher level cache for writing back modified data. However, it is well

known in the art, when flushing a lower level cache, to write back modified data to higher levels of cache and to the main memory, [which effectively couples the higher level cache and the main memory to the flush mechanism], as disclosed by the Applicant in the background of the invention (page 3, line 20-21; page 4, lines 1-6) for the purpose of ensuring coherency. Therefore, it would have been obvious to one of ordinary skill in the art to write back modified data to a higher level cache, thereby coupling a cache flush mechanism to the higher level cache, for the desirable purpose of coherency.

Regarding claims 11-12, McDermott and Masubuchi disclose the cache flush mechanism logically coupled to the main memory for writing back modified data (McDermott - C 8, L 22-62; C 9, L 10-16 - the dirty cache lines are written back to memory during a flush operation; the lines are invalidated which effectively evicts the cache lines since an access to the invalidated cache lines will cause a cache miss).

Regarding claim 14, McDermott and Masubuchi disclose the list structure located on a die (McDermott - Figure 4, Reference 331, Figure 2, Reference 200, 204 – the list structure is comprised within the processor and thus it is evident that the list structure is located on a die comprising the processor).

Regarding claim 16, McDermott discloses a computer system (Figure 1) with a cache memory (Figure 2, Reference 204) and an apparatus for flushing the cache (Figure 4, Reference 330; C 8, L 64-67) comprising a list structure (Figure 4, comprised of References 331, 332) for recording

modifications to a plurality of cache entries wherein the list structure does not contain cache data or addresses (C 3, L 3-7; C 9, L 1-17, L 56-62; C 10, L 1-2); a cache controller (Figure 3a, Reference 303) adapted to query the list structure [via scanning logic, Reference 334 in Figure 4] for modifications to the plurality of cache entries and generate a list (series of) of cache write-back instructions (C 10, L 12-32; C 9, L 10-17 – when a flush operation is initiated, the list structure is scanned to identify dirty data, each identified dirty data is written-back to memory intrinsically via a write-back command); and wherein the cache controller invalidates the plurality of cache entries corresponding to the list of cache write-back instructions (C 10, L 11-32; C 9, L 10-17 – when a flush operation occurs clean data is invalidated [intrinsically via an invalidate command] and dirty data is written back to memory and thereafter invalidated [intrinsically via an invalidate command]). McDermott does not explicitly disclose the list structure located outside the cache. However, Masubuchi discloses a list structure located outside of a cache (Figure 1, the list structure is comprised of References 32, 33, and 34). This feature taught by Masubuchi reduces logic in the cache, which provides space for other/additional logic or otherwise allows the size of the cache to be reduced. Additionally, this feature relieves the cache from processing cache flushes, thereby allowing the cache to perform other tasks. Hence, it would have been obvious to one of ordinary skill in the art to locate the list structure in McDermott's system outside of the cache to reduce the logic in the cache and to allow external processing of the list structure to alleviate processing operations of the cache.

Regarding claims 17 and 19, McDermott discloses the list structure as a full list, comprising one entry per cache line (Figure 3, Reference 331 - there is one entry per cache line for each cache line represented in the list structure).

Regarding claims 18 and 20, McDermott discloses the list structure as a partial list comprising one entry per plurality of cache lines (plurality of cache lines per way/set)(McDermott - Figure 4, Reference 332; C 9, L 35-37).

Regarding claims 25, McDermott discloses creating a table of cache entries without the cache data or addresses (Figure 4, Reference 331- the table is created by setting or unsetting the NDIRTY bits for the cache lines); tracking (via scanning logic) modified cache entries in the table (C 9, L 33-35; C 10, L 11-32); and generating a write-back command [generated when performing a copy back operation of dirty data] from the table in response to a cache flush event (C 10, L 12-32; C 9, L 10-17 – when a flush operation is initiated, the table is scanned to identify dirty data, wherein the dirty data is exported [copied back to memory] and invalidated). McDermott does not disclose the table separate from the cache. However, Masubuchi discloses a table separate from a cache for tracking modified cache entries (Figure 1, the table is comprised of References 32, 33, and 34). This feature taught by Masubuchi reduces logic in the cache, which provides space for other/additional logic or otherwise allows the size of the cache to be reduced. Additionally, this feature relieves the cache from processing cache flushes, thereby allowing the cache to perform other tasks. Hence, it would have been obvious to one of ordinary skill in the art to locate the list structure in McDermott's system outside of the cache to

reduce the logic in the cache and to allow external processing of the list structure to alleviate processing operations of the cache.

Regarding claim 26, McDermott and Masubuchi disclose generating an invalidate command in response to the cache flush (McDermott – C 10, L 11-32; C 9, L 10-17 – when a flush operation occurs clean data is invalidated [intrinsically via an invalidate command] and dirty data is written back to memory and thereafter invalidated [intrinsically via an invalidate command]).

Regarding claim 27, McDermott and Masubuchi disclose performing the method of claim 25 for a L1 cache. McDermott and Masubuchi do not disclose performing the method for each level cache. However, McDermott suggest the idea of performing the method also for a L2 cache for the purpose of speeding cache flushes (C 11, L 21-24). The system taught by McDermott and Masubuchi comprises a L2 cache (McDermott - Figure 1, Reference 404). Hence, it would have been obvious to one of ordinary skill in the art to repeat the method of claim 25 for the L2 cache in the system taught by McDermott and Masubuchi for the desirable purpose of speeding cache flushes thereby improving the performance of the system.

Regarding claim 29, McDermott and Masubuchi disclose writing-back modified entries to memory (McDermott - C 8, L 22-62; C 9, L 10-16 - the dirty cache lines are written back to memory during a flush operation).

Regarding claim 30, McDermott and Masubuchi do not disclose the cache flush mechanism logically coupled to a higher level cache for writing back modified data. However, it is well known in the art, when flushing a lower level cache, to write back modified data to higher levels of cache and to the main memory, [which effectively couples the higher level cache and the main memory to the flush mechanism], as disclosed by the Applicant in the background of the invention (page 3, line 20-21; page 4, lines 1-6) for the purpose of ensuring coherency. Therefore, it would have been obvious to one of ordinary skill in the art to write back modified data to a higher level cache, thereby coupling a cache flush mechanism to the higher level cache, for the desirable purpose of coherency.

4. Claims 5-6 are rejected under 35 U.S.C. 103(a) as being unpatentable over McDermott (USPN: 5,860,105) in view of Masubuchi et al. (USPN: 6,490,657) as applied to claim 1 above and further in view of Arimilli (USPN: 6,058,456).

McDermott and Masubuchi do not disclose the apparatus comprising one bit per a variable number of cache lines (claim 5), which is set by the operating system (claim 6) and wherein the logical arrangement of the list structure conforms to the variable number (claim 5). However, Arimilli teaches the concept of an operating system programmably selecting an associativity level for a cache, which dictates the number of cache lines per set/way (C 6, L 22-28). Arimilli teaches that this feature allows the cache to operate more efficiently (C 5, L 9-10). In the system taught by McDermott and Masabuchi, the list structure array comprises one bit per set, wherein each set comprises a plurality of cache lines and thus the structure comprises one bit per plurality of cache lines (Figure 4, Reference 332). In modifying the system taught by McDermott and

Masubuchi, with the teachings of Arimilli, the number of cache lines per set would vary depending on the selected associativity for the cache and thus the number of cache lines per bit would also vary. Additionally, it would be desirable to use the teachings of Arimilli with the system taught by McDermott and Masubuchi for the purpose of efficiency. Hence, it would have been obvious to one of ordinary skill in the art to use the teachings of Arimilli with the system taught by McDermott and Masubuchi, [wherein the list structure conforms to the variable number of cache lines by providing one bit per set, wherein each set comprises a variable number of cache lines], for the desirable purpose of efficiency.

5. Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over McDermott (USPN: 5,860,105) in view of Masubuchi (USPN: 6,490,657) as applied to claim 1 above and further in view of Otterness et al. (USPN: 6,460,122).

Regarding claim 13, McDermott and Masubuchi do not explicitly disclose the list structure located within a RAM, however, Otterness discloses a list structure located in a RAM (Figure 3, Reference 204). RAMs [particularly, SRAMs] offer fast access to data stored therein and thus provides fast retrieval of data. Hence, it would have been obvious to one of ordinary skill in the art to locate the list structure taught by McDermott and Masubuchi in a RAM for the desirable purpose of fast retrieval of data.

6. Claims 15 and 28 is rejected under 35 U.S.C. 103(a) as being unpatentable over McDermott (USPN: 5,860,105) in view of Masubuchi (USPN: 6,490,657) as applied to claims 1 and 25 and further in view of Stevens (USPN: 5,724,550).

McDermott and Masubuchi disclose the limitations cited above in claims 1 and 25, however, McDermott and Masubuchi do not disclose querying the list structure in response to a snoop command. However, Stevens teaches the concept of querying a list structure (structure which indicates the state, such as modified/dirty, of a cache line) to determine the status of a cache line in response to a snoop command (C 7, L 32-44) to ensure the correct copy of data is retrieved by a requesting device (C 1, L 49-67; C 2, L 1). Hence, it would have been obvious to one of ordinary skill in the art to use Stevens' teachings with the system taught by McDermott and Masubuchi, in an environment providing shared access to cache lines, for the desirable purpose of accuracy.

7. Claims 21-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Masubuchi (USPN: 6,490,657) in view of McDermott (USPN: 5,860,105).

Regarding claims 21 and 24, Masubuchi discloses a multiprocessor computer system with a plurality of processors and caches and with an apparatus for cache flushing (Figure 1) comprising a list structure for tracking the status of a plurality of cache entries, wherein the list structure is located outside the cache (Figure 1, Reference 32; C 11, L 17-19, L 41-53; C 11, L 60-67; C 12, L 1-5); a processor identification within the list structure for linking the plurality of cache entries to one of the plurality of processors (C 11, L 41-67; C 12, L 1-5; C 13, L 9-13; C 14, L 13-23); a query mechanism for checking the list structure for the state of a cache entry identified with a processor (C 12, L 13-30 - logic within Reference 5 of Figure 1 which performs checking); a cache flush mechanism for flushing a cache entry linked to an identified processor and for modifying the list structure to reflect the flushed state (C 12, L 13-30, L 33-34 – logic

within Reference 32 which performs cache flushing and modifying). Masubuchi does not disclose the list structure not containing cache data or addresses and located on a die with at least one of the plurality of processors (claim 24). However, McDermott discloses a list structure that does not contain cache data or addresses (Figure 4, comprised of References 331, 332) and located on a die of a processor (the list structure is comprised within the processor and thus it is evident that the list structure is located on a die comprising the processor). The list structure taught by McDermott tracks the status of a plurality of cache entries using minimal information thereby providing a smaller and faster structure [faster due to reduced delays provided by having less internal logic, wiring, etc.], in comparison to the list structure taught by Masubuchi, and provides fast cache flushing using a lookahead scheme. Hence, one of ordinary skill in the art would have been motivated to use the teachings of McDermott in the system taught by Masubuchi for the desirable purpose of space efficiency and fast cache flushing.

Regarding claim 22, the system taught by Masubuchi and McDermott disclose the list structure comprising one bit per cache line (Masubuchi - C 14, L 13-16; Configuration 3-A; C 17, L 21-24 – one of the entry bits for each region represents one bit per cache line).

Regarding claim 23, the system taught by Masubuchi and McDermott disclose the list structure comprising one bit per plurality of cache lines (Masubuchi - Configuration 4-A; C 17, L 52-59; one of the bits in each Region represents one bit per plurality of cache lines).

Response to Arguments

8. Applicant's arguments filed have been fully considered but they are not persuasive. The previous responses to Applicant's arguments have been restated as Applicant's present arguments are the same as previously stated.

Regarding Applicant's argument regarding the teachings of Masabuchi, it should be noted that Masabuchi is relied upon for teaching that which McDermott does not. McDermott teaches a list structure which does not contain cache data or addresses. McDermott does not teach the list structure located outside of a cache. Masabuchi is relied upon for teaching a list structure which is located outside of a cache. Hence, it is not clear why the Applicant is arguing that Masabuchi's list structure contains addresses, since Masabuchi was never relied upon for teaching that feature.

Regarding Applicant's arguments concerning the motivation for combining the teachings of Masabuchi with the teachings of McDermott, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, **or motivation to do so found** either in the references themselves or **in the knowledge generally available to one of ordinary skill in the art.** See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, it is knowledge generally available to one of ordinary skill in the art that removing logic or elements from a device reduces the amount of logic in device and allows other logic to substitute the removed logic or the device may be reduced in size by the amount of logic removed. It is also within the knowledge generally

available to one of ordinary skill in the art that by moving logic from one place to another, the device no longer has to execute the functionality of that logic because the logic has been removed. In doing so, the device is relieved of processing certain functions.

Additionally, it should be noted that a reference does not teach away from a particular modification merely because it is silent to making the modification. Just because McDermott teaches a list structure within the cache does not necessarily infer that McDermott teaches away from having the list structure located outside of the cache.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kimberly N. McLean-Mayo whose telephone number is 703-308-9592. The examiner can normally be reached on M-F (9:00 - 6:30) First Friday Off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald Sparks can be reached on 703-308-1756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

KIMBERLY MCLEAN-MAYO
PRIMARY EXAMINER

Kimberly N. McLean-Mayo
Examiner
Art Unit 2187

KNM

May 23, 2004